APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/887,561	06/25/2001	Edward Colles Nevill	550-244	7551

7590 05/02/2005	**EXAMINER**
NIXON & VANDERHYE P.C.	TANG, KUO LIANG J

NIXON & VANDERHYE P.C.
8th Floor
1100 North Glebe Road
Arlington, VA 22201

ART UNIT	PAPER NUMBER
2191	

DATE MAILED: 05/02/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

	Office Action Summary	**Application No.**	**Applicant(s)**
		09/887,561	NEVILL, EDWARD COLLES
		Examiner	**Art Unit**
		Kuo-Liang J Tang	2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>24 November 2004</u>.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
  closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-16 and 20-28* is/are pending in the application.

  4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-16 and 20-28* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

  Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

  Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

  a)☐ All   b)☐ Some * c)☐ None of:

  1.☐ Certified copies of the priority documents have been received.

  2.☐ Certified copies of the priority documents have been received in Application No. _____.

  3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
    application from the International Bureau (PCT Rule 17.2(a)).

  * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
  Paper No(s)/Mail Date _____.
4)☐ Interview Summary (PTO-413)
  Paper No(s)/Mail Date. _____ .
5)☐ Notice of Informal Patent Application (PTO-152)
6)☐ Other: _____.

DETAILED ACTION

1.   This Office Action is in response to the amendment filed on 11/24/2004.

The priority date for this application is 10/05/2000.

Response to Arguments

2.   Applicant's arguments with respect to Claims 1-16 and 20-28 have been considered but

they are not persuasive.

Claims 20-21 have been amended.

Claims 1-16 and 20-28 are pending and have been examined.

Claims 1-5, 15-16 and 20 remain rejected under 35 U.S.C. 102(b) as being anticipated by

Guccione, "Portable Native Methods in Java".

Claims 6-9 and 21-24 remain rejected under 35 U.S.C. 103(a) as being unpatentable over

Guccione in view of Yates et al. US Patent No. 6,091,897 (hereinafter Yates).

Claims 10-14 and 25-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Guccione in view of Weiss, "Introduction to Native Calls".

In the Amendment, Applicant argues that:

A)   As for independent claims 1, 15 and 20, Applicant primarily argues that

Guccione's Figure 5 is an instruction in the native (C) programming language and not a return

instruction in the non-native (Java) programming language. (see Amendment page 7, lines 8-

20).

B)   Applicant primarily argues that DLL is not an instruction translator. (see Amendment page 8, lines 10-13).

Examiner's response:

A)   The examiner disagrees with Applicant's assertion that Figure 5 is not a return instruction in the non-native (Java) programming language. In fact, Guccione's Figure 5 is the Java stubs interface code, therefore, the return is a return instruction in the non-native (Java) programming language. Further, as noted in the previous Office Action (dated 08/02/2004), page 3, last paragraph, Guccione shows teaching of translator that is responsive to a return to non-native instruction (E.g. Java code) to return processing to a non-native instruction (E.g. Java code) using stubs (E.g. C and Java code).

B)   The examiner disagrees with Applicant's assertion that DLL is not an instruction translator. In fact, as Applicant admitted that "The stubs code is intend to enable a native method to be incorporated in a non-native computer program" (See Amendment page 8, lines 5-6). Therefore, after compiling the stub code into a DLL, this DLL performs the same functionality like Claim 1 (iv).

Claim Rejections - 35 USC § 102

3.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4.   Claims 1-5, 15-16 and 20 are rejected under 35 U.S.C. 102(b) as being anticipated by

Guccione, "Portable Native Methods in Java".

As Per Claim 1, Applicant's admitted prior art discloses the method that covering the

steps of:

"(i) a processor core operable to execute native instructions (E.g. C code) of a native

instruction set;"

"(ii) an instruction translator (E.g. see page 3, Figure 1, "Stubs DLL" and associated

text) operable to interpret non-native instructions (E.g. Java code) of a non-native instruction set

into native instructions (E.g. C code) for execution by said processor core;"

"(iii) said instruction translator (E.g. see page 3, Figure 1, "Stubs DLL", Figure 2, 4-5

and associated text, e.g. mix of C and java codes) is responsive to a return to non-native

instruction (E.g. see page 5, Figure 5 and associated text) of said non-native instruction set to

return processing to a non-native instruction (E.g. Java code);" and

"(iv) said instruction translator is responsive to a return to native instruction (E.g. C code)

of said non-native instruction set to return processing to a native instruction (E.g. C code)."

Although Guccione only shows teaching of translator that is responsive to a return to

non-native instruction (E.g. Java code) to return processing to a non-native instruction (E.g. Java

code) using stubs (E.g. C and Java code). The Stubs is created for both directions so the native

instruction and non-native instruction can exchange some information. Therefore, the opposite

direction (translator is responsive to a return to native instruction of said non-native instruction

set to return processing to a native instruction) must be disclosed otherwise the stubs would not

work at all.

As per Claims 2, the rejection of claim 1 is incorporated and further Guccione teaches:

"said instruction translator is a hardware based instruction translator." (E.g., see page 3,

line 1, "The examples below were developed on an IBM-PC compatible machine running

Microsoft Windows NT/4.0.").

As per Claims 3, the rejection of claim 1 is incorporated and further Guccione teaches:

"said instruction translator is a software based interpreter." (E.g. see page 4, line 7, "Sun

Java 1.0 interpreter").

As per Claims 4, the rejection of claim 1 is incorporated and further Guccione teaches:

"said instruction translator is a combination of a hardware based instruction translator and

a software based interpreter." (Again, see as noted above of Claims 2-3).

As per Claims 5, the rejection of claim 1 is incorporated and further Guccione teaches:

"said non-native instructions are Java Virtual Machine instructions." (E.g. see page 1,

ABSTRACT).

As Per Claim 15, is the method claim corresponding to the apparatus claim 1 and is

rejected under the same reason set forth in connection of the rejection of claim 1.

As Per Claim 16, is the computer program product claim corresponding to the apparatus

claim 15 and is rejected under the same reason set forth in connection of the rejection of claim

15.

As per Claims 20, Guccione teaches:

"processing means for executing native instructions (E.g. C code) of a native instruction

set;"

"translator means for interpreting non-native instructions (E.g. Java code) of a non-

native instruction set into native instructions (E.g. C code) for execution by said processor core

being responsive to a return to non-native instruction (E.g. see page 5, Figure 5 and associated

text) of said non-native instruction set to return processing to a non-native instruction (E.g. Java

code), and responsive to a return to native instruction (E.g. C code) of said non-native instruction

set to return processing to a native instruction (E.g. C code)."

Although Guccione only shows teaching of translator that is responsive to a return to

non-native instruction (E.g. Java code) to return processing to a non-native instruction (E.g. Java

code) using stubs (E.g. C and Java code). The Stubs is created for both directions so the native

instruction and non-native instruction can exchange some information. Therefore, the opposite

direction (translator is responsive to a return to native instruction of said non-native instruction

set to return processing to a native instruction) must be disclosed otherwise the stubs would not

work at all.

5.   Claims 6-9 and 21-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Guccione in view of Yates et al. US Patent No. 6,091,897 (hereinafter Yates).

As Per Claim 6, the rejection of claim 1 is incorporated and further Guccione doesn't

explicitly disclose a non-native veneer subroutine. However, Yates teaches "a non-native

subroutine is called from native code via a non-native veneer subroutine, such that, upon

completion of said non-native subroutine, a return to non-native instruction can be used to return

processing to said non-native veneer subroutine with a return to native instruction within said

non-native veneer subroutine serving to return processing to said native code" (E.g., see col.

33:18-36; "translated" is native, "not translated" is non-native). Therefore, it would have been

obvious to one of ordinary skill in the art at the time the invention was made to incorporate the

teaching of Yates into the system of Guccione, to use a non-native veneer subroutine. The

modification would have been obvious because one of ordinary skill in the art would have been

motivated to invoke a <u>subroutine</u> call to an appropriate <u>native subroutine</u> when said instruction

fetch is for a complex <u>non-native</u> instruction.

As Per Claim 7, the rejection of claim 6 is incorporated and further Guccione doesn't

explicitly disclose non-native subroutine is also called from non-native code. However, Yates

teaches "non-native subroutine is also called from non-native code." (E.g., see col. 33:18-36).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention

was made to incorporate the teaching of Yates into the system of Guccione, to use non-native

subroutine called from non-native code. The modification would have been obvious because one

of ordinary skill in the art would have been motivated to invoke a type of subroutine with same

type of code instructions set to keep the program code unique and portable.

As Per Claim 8, the rejection of claim 6 is incorporated and further Guccione doesn't

explicitly disclose non-native veneer subroutine is dynamically created. However, Yates teaches

"non-native veneer subroutine is dynamically created when said non-native subroutine is called

from native code." (E.g., see col. 33:18-36). Therefore, it would have been obvious to one of

ordinary skill in the art at the time the invention was made to incorporate the teaching of Yates

into the system of Guccione, to use dynamically created non-native veneer subroutine. The

modification would have been obvious because one of ordinary skill in the art would have been

motivated to use translated routine calls another translated Routine.

As Per Claim 9, the rejection of claim 8 is incorporated and further Guccione doesn't

explicitly disclose non-native veneer subroutine is created stored within a stack memory area

used by native code operation. However, Yates teaches "non-native veneer subroutine is created

stored within a stack memory area used by native code operation." (E.g., see col. 33:18-36).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention

was made to incorporate the teaching of Yates into the system of Guccione, to use dynamically

created non-native veneer subroutine. The modification would have been obvious because one of

ordinary skill in the art would have been motivated to use translated routine calls another

translated Routine.

As per Claims 21-24, the rejection of claim 1 is incorporated and is rejected under the same reason set forth in connection of the rejection of claims 6-9.

6.   Claims 10-14 and 25-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Guccione in view of Weiss, "Introduction to Native Calls".

As Per Claim 10, the rejection of claim 1 is incorporated and further Guccione doesn't explicitly disclose plurality of types of return. However, Weiss teaches "said instruction translator is responsive to a plurality of types of return to non-native instruction." (E.g., see pages 5-6, slides 13-18). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Weiss into the system of Guccione, to be responsive to a plurality of types of return to non-native instruction. The modification would have been obvious because one of ordinary skill in the art would have been motivated to create strings and arrays in the stub when the native method has parameters, return types, exceptions.

As Per Claim 11, the rejection of claim 10 is incorporated and further Guccione doesn't explicitly disclose said plurality of types of return to non-native instruction are operable to return with respective different types of return value. However, Weiss teaches "said plurality of types of return to non-native instruction are operable to return with respective different types of return value." (E.g., see pages 5-6, slides 13-18). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Weiss

into the system of Guccione, to operable to return with respective different types of return value.

The modification would have been obvious because one of ordinary skill in the art would have

been motivated to create strings and arrays in the stub when the native method has parameters,

return types, exceptions.

As per Claims 12, the rejection of claim 11 is incorporated and further the combination of

Guccione and Weiss teaches:

"said plurality of different types of return value include one of: a 32-bit integer return

value ..." (E.g., see Weiss, page 5, slide 13).

As Per Claim 13, the rejection of claim 1 is incorporated and further Guccione doesn't

explicitly disclose plurality of types of return. However, Weiss teaches "said instruction

translator is responsive to a plurality of types of return to native instruction." (E.g., see page 9,

slide 27). Therefore, it would have been obvious to one of ordinary skill in the art at the time the

invention was made to incorporate the teaching of Weiss into the system of Guccione, to be

responsive to a plurality of types of return to native instruction. The modification would have

been obvious because one of ordinary skill in the art would have been motivated to create strings

and arrays in the stub when the native method has parameters, return types, exceptions.

As Per Claim 14, the rejection of claim 13 is incorporated and further Guccione doesn't

explicitly disclose said plurality of types of return to native instruction are operable to return

with respective different types of return value. However, Weiss teaches "said plurality of types

of return to native instruction are operable to return with respective different types of return value." (E.g., see page 9, slide 27). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Weiss into the system of Guccione, to operable to return with respective different types of return value. The modification would have been obvious because one of ordinary skill in the art would have been motivated to create strings and arrays in the stub when the native method has parameters, return types, exceptions.

As per Claims 25-28, the rejection of claim 20 is incorporated and is rejected under the same reason set forth in connection of the rejection of claims 10-11, 13-14.

Conclusion

7.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Correspondence Information

8.   Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Kuo-Liang J Tang whose telephone number is (571) 272-3705.

The examiner can normally be reached on 8:30AM - 7:00PM (Monday – Thursday). Any

inquiry of a general nature or relating to the status of this application should be directed to the

TC 2100 Group receptionist: 571-272-2100.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Tuan Dam can be reached on (571) 272-3695. The fax phone number for the

organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Software Engineer Patent Examiner

ANTONY NGUYEN-BA
PRIMARY EXAMINER